

INSIDE DOS™

Tips & techniques for MS-DOS & PC-DOS

Viewing the contents of a text file in DOS 5's Shell

When you need to view the contents of a text file in DOS, you probably take advantage of the system prompt's TYPE command. For example, if you need to determine the contents of your AUTOEXEC.BAT file, you can type at the root directory the command

```
C:\>type autoexec.bat
```

When you do, DOS will display on the screen all the commands you've placed in AUTOEXEC.BAT. If you don't need to modify the file, viewing it with the TYPE command is much quicker than loading it into a text editor and viewing it there.

Not surprisingly, Microsoft built this same capability into the DOS 5 Shell. If you're working in the Shell and you want to take a look at a file's contents, you can use the File menu's View File Contents command or simply press the [F9] key. When you do, DOS will display the contents of the file in a window on your screen. If the file contains more text than will fit in one screen, you can use the [Page Up], [Page Down], ↓, and ↑ keys to scroll through the file. To exit the View window and return to the Shell screen, press [Esc].

As an example of viewing a text file in the Shell, let's take a look at DOS 5's README.TXT file. This file, which is stored in the \DOS directory, contains information about DOS 5 that isn't included in the manual.

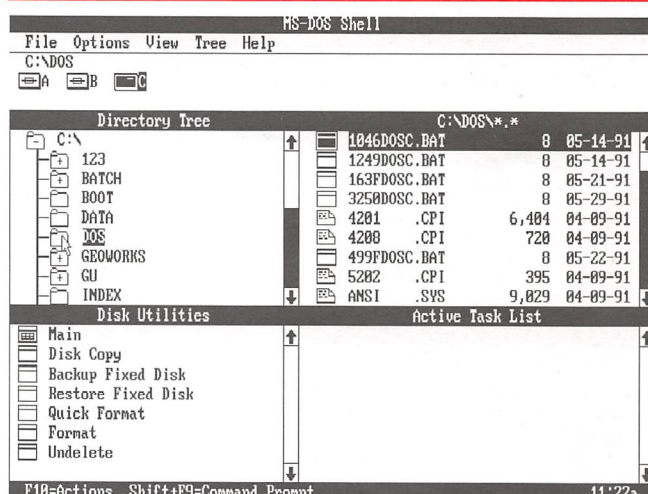
If you haven't loaded the Shell program, do so by typing

```
C:\>dosshell
```

Next, select the \DOS directory by clicking with your mouse on its icon in the Directory Tree window. Or, to select the \DOS directory from the keyboard, press [Tab] once to activate the Directory Tree window, then press the ↓ key until you select the \DOS directory. When you select the \DOS directory, its files will appear in the File List window, as shown in Figure A.

Before you can issue the View File Contents command, you need to tell DOS which file you want to look at by selecting its filename in the File List window. To activate the File List window, click in it with your mouse, or press the [Tab] key.

Figure A



When you select a directory in the Shell, its contents appear in the File List window.

If you don't know the filename, you can scroll through the list of files by clicking on the up or down arrows that appear on the right side of the File List window, then using the cursor movement keys (↓, ↑) or the mouse to select the file. However, when you know the name of the file, you can quickly scroll to its general vicinity by pressing the key that corresponds to the first letter of the filename. Since you want to highlight the file README.TXT, press R.

Continued on page 12

IN THIS ISSUE

- Viewing the contents of a text file in DOS 5's Shell 1
- You may already be a WINner 2
- Searching for files by date in DOS 5 3
- A second chance: How DOS 5 "undeletes" files 5
- Protecting your disk with DOS 5's MIRROR program 7
- Getting to know the DOS environment 8
- Van Wolverton: Looking into the crystal ball: What lies ahead for the computer industry in 1992 9

INSIDE DOS

Inside DOS (ISSN 1049-5320) is published monthly by The Cobb Group.

Prices	Domestic	\$59/yr. (\$7.00 each)
	Outside the US	\$79/yr. (\$8.50 each)
Address	The Cobb Group	
	9420 Bunsen Parkway, Suite 300 Louisville, KY 40220	
Phone	Toll-free	(800) 223-8720
	Local	(502) 491-1900
	FAX	(502) 491-4200
Staff	Editor-in-Chief	Jim Welp
	Consulting Editors	Mark W. Crane
		Tim Landgrave
	Contributing Editor	Van Wolverton
	Editing	Clyde Zellers
		Polly Blakemore
	Production	Debbie Lane
	Design	Karl Feige
	Publications Manager	Elayne Noltemeyer
	Publisher	Douglas Cobb

Address correspondence and special requests to The Editor, *Inside DOS*, at the address above. Address subscriptions, fulfillment questions, and requests for bulk orders to Customer Relations, at the address above.

Postmaster: Send address changes to *Inside DOS*, P.O. Box 35160, Louisville, KY 40232. Second class postage is paid in Louisville, KY.

Copyright © 1992, The Cobb Group. All rights reserved. *Inside DOS* is an independently produced publication of The Cobb Group. No part of this journal may be used or reproduced in any fashion (except in brief quotations used in critical articles and reviews) without prior consent of The Cobb Group.

The Cobb Group, its logo, and the Satisfaction Guaranteed statement and seal are registered trademarks of The Cobb Group. *Inside DOS* is a trademark of The Cobb Group. Microsoft is a registered trademark of Microsoft Corporation. IBM is a registered trademark of International Business Machines, Corporation.

Conventions

To avoid confusion, we would like to explain a few of the conventions used in *Inside DOS*.

When we instruct you to type something, those characters usually appear on a separate line along with the DOS prompt. The characters you type will appear in red, while the characters DOS displays will appear in black.

Occasionally, we won't display the command you type on a separate line. In these cases, we'll display the characters you type in italics. For example, we might say, "Issue the command *dir *.txt* at the DOS prompt." Although DOS is not case-sensitive, we'll always display the characters you type in lowercase.

When we refer to a general DOS command (not the command you actually type at the DOS prompt), we'll display that command name in all caps. For example, we might say, "You can use either the COPY or XCOPY command to transfer files from one disk to another."

Many commands accept parameters that specify a particular file, disk drive, or other option. When we show you the form of such a command, its parameters will appear in italics. For example, the form of the COPY command is

copy file1 file2

where *file1* and *file2* represent the names of the source file and the target file, respectively.

The names of keys, such as [Shift], [Ctrl], and [F1], appear in brackets. When two keys must be pressed simultaneously, those key names appear side by side, as in [Ctrl][Break] or [Ctrl]z.

LETTERS

You may already be a WINner

I recently upgraded to DOS 5. One of the files listed in my root directory is WINA20.386. I've tried everything I can think of to access this file but have been unsuccessful.

Can you tell me what WINA20.386 is (it appears to be some sort of contest through Microsoft) and how I can access it?

William D. Casey
Independence, Kansas

We'd like to report that Microsoft is sponsoring a contest in which the first person to figure out the purpose for WINA20.386 wins a 20 megahertz 386 personal computer. Unfortunately, the truth isn't so exciting.

The WINA20.386 file is a read-only file that the MS-DOS Setup program copies to your disk's root directory if your system is capable of running Microsoft Windows 3.0 in Enhanced mode. This file allows Windows to switch between virtual DOS machines when running in Enhanced mode. As you might expect, Microsoft Windows 3.0 requires that this file be present in order to run in Enhanced mode. If you attempt to start Windows 3.0 in Enhanced mode, and Windows cannot find the file WINA20.386, you will see the message:

You must have the file WINA20.386 in the root directory of your boot drive to run Windows in Enhanced mode.

Deleting WINA20.386

If you want to remove the file WINA20.386 from your root directory, you have two options. First, if you never run (or never intend to run) Microsoft Windows 3.0 in Enhanced mode, you can delete WINA20.386. In order to delete WINA20.386 you first need to turn off the file's read-only attribute. To do this, issue the command:

```
C:\>attrib -r c:\wina20.386
```

Once you've turned off the file's read-only attribute, delete it in the usual fashion with the command:

```
C:\>del c:\wina20.386
```

Moving WINA20.386

If you want to remove WINA20.386 from your root directory, but still plan to run Windows 3.0 in Enhanced mode, you can move the file out of your root directory and into a subdirectory. To do this, follow these steps:

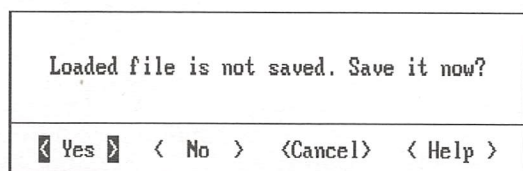
1. Use the DOS Editor to modify your CONFIG.SYS file by typing

```
C:\>edit c:\config.sys
```

2. Near the bottom of your CONFIG.SYS file add the line

`switches=/w`
3. Exit the DOS Editor and save your changes by choosing the Exit command from the File menu. When the DOS Editor displays the dialog box shown in Figure A, choose Yes or press [Enter].

Figure A



The DOS 5 Editor asks you if you want to save your changes before you exit the program.

4. Use Edit to open the Windows SYSTEM.INI file (this file should appear in your Windows directory). For instance, if you've installed Windows in the directory C:\WINDOWS, type

```
C:\>edit c:\windows\system.ini
```

5. Locate the section of SYSTEM.INI that begins with the header [386Enh].

6. Under the [386Enh] heading add the following command:

```
device=path
```

where *path* is the new fully qualified path name of the file WINA20.386. For example, if you move WINA20.386 into the directory C:\WINDOWS, you would enter into the [386Enh] section of SYSTEM.INI the line

```
device=c:\windows\wina20.386
```

7. Save the changes to your SYSTEM.INI file by selecting the Exit command from the File menu, and then choosing Yes in the ensuing dialog box.
8. Once you've made the appropriate changes to CONFIG.SYS and SYSTEM.INI, restart your system by pressing [Ctrl][Alt][Del].

Searching for files by date in DOS 5

I'm looking for a means of finding and listing all the files on my hard drive that I created on a particular date. I tried issuing the command

```
dir | find "MM-DD-YY"
```

where *MM-DD-YY* is the date I created the files, but when I press [Enter] DOS displays only the files in the current directory.

DOS 5's DIR command

```
C:\>dir /od /s
```

lists *all* the files on my hard disk, sorted by date. Is there a command in DOS 5 I can use to list only the files created on a given date?

*Allen Sven Oxenburgh
New York, New York*

As we've shown in recent issues of *Inside DOS*, DOS 5's new and improved DIR command makes it easier to sort files than in previous versions. For instance, one new switch, /S, generates a listing of all the files in all the subdirectories on your hard drive. To display every file on your hard drive, issue the command

```
C:\>dir /s
```

To display the files one screen at a time, employ the /P switch:

```
C:\>dir /s /p
```

If you want to see all the files that match a certain file specification, you can use DOS wildcards. For example, to see all the files that contain the extension TXT, issue the command

```
C:\>dir /s *.txt
```

As you'll see in a moment, the /S switch is instrumental in generating a list of files created on a certain date. Let's take a quick look at how you can sort files by date, then we'll show you a command that displays only the files created on a specific date.

Sorting a directory listing

DOS 5 includes another new switch, /O, that lets you sort files at the command prompt. Using the /O switch you can sort files by name, date, extension, and size. (For more on sorting files with DOS 5's DIR command, see the article "Using DOS 5's DIR Command to Sort Files" in the July 1991 issue of *Inside DOS*.)

Sorting files by date

To sort files in the current directory by date with the earliest entry first, use the /OD switch:

```
C:\ISLAND>dir /od
```

```
Volume in drive C is DOS
Volume Serial Number is 2D10-13F5
Directory of C:\ISLAND

.                <DIR>      03-06-91    4:41p
..               <DIR>      03-06-91    4:41p
PROF            DOC      29268 03-10-91    8:46p
MARYANN         DOC      31833 04-16-91    8:25a
GINGER          DOC      11219 05-12-91   11:09a
SKIPPER         DOC      12321 10-03-91   10:10a
MRHOWELL        DOC      22437 10-03-91   11:15p
7 file(s)        107078 bytes
196210688 bytes free
```

If you'd like to generate a list of all the files on your hard drive sorted by date, at the root directory add the /S switch like this:

```
C:\>dir /od /s
```

When you press [Enter], DOS will list all the files in each directory according to the date they were created or last modified. To see the list one screen at a time, again add the /P switch:

```
C:\>dir /od /s /p
```

Limiting the listing to a certain date

Of course, listing all the files on your hard drive can be a cumbersome way to find files created on a certain date. For that purpose, you're better off using the FIND command. As you may know, the FIND command searches for the text you specify in quotation marks. Since every item listed in a directory includes a date, DOS' search returns only the files that contain the date you specify in quotation marks. Also, because you want DOS to look in every subdirectory on your hard drive, you need to include the /S switch. Therefore, the form of the command you need to use is

```
C:\>dir /s | find "MM-DD-YY"
```

where *MM-DD-YY* is the date portion of the filenames for which you're searching.

For example, to find all the files created or last modified on 10-03-91, you'd issue the command

```
C:\>dir /s | find "10-03-91"
```

To see the listing one screen at a time, append the MORE filter to the command:

```
C:\>dir /s | find "10-03-91" | more
```

Automating the process

If you'd like to be able to search for files by date frequently, but don't want to memorize the syntax, you can store the above command in a Doskey macro, specifying a replaceable parameter as the date. To create a DATELIST macro, type

```
C:\>doskey datelist=dir /s %b find "%1" %b more
```

Once you create the DATELIST macro, you can type its name, followed by the date of the files you want DOS to find. For example, to find files created or last modified on 10-03-91, issue the command

```
C:\>datelist 10-03-91
```

As you can see, there are a couple of differences between the command-line version of our command and the macro version. First, Doskey's macro language calls for the characters *%b* instead of the pipe character (*|*). Also, because you want to be able to specify any date, you'll need to use the replaceable parameter *%1* in the macro definition. Doing so tells DOS to replace the parameter *%1* with whatever you type after the macro's name in the command. In our case, DOS replaces the parameter *%1* with whatever date you type after *datelist*. (For the ins and outs of DOS 5's new macro feature, see the article "Doskey Macros put Complex Commands at Your Fingertips" in the September 1991 issue of *Inside DOS*)

Locating the files

Unfortunately, when you issue the above command, DOS simply lists all the files created on the date you specified—it doesn't supply the location of the file. For instance, suppose on 10-03-91 you created two files named ELVIS.DOC that you stored in different subdirectories. When you issue the command

```
C:\>dir /s | find "10-03-91"
```

DOS will respond

```
ELVIS          DOC      10-03-91    6:41p
ELVIS          DOC      10-03-91    5:25p
```

Once you know the name of the file you're searching for, you can generate a listing that includes the pathname by using DIR's /B switch in combination with the /S switch, like this:

```
C:\>dir elvis.doc /s /b
```

DOS will respond

```
C:\COSTELLO\ELVIS.DOC
C:\PRESLEY\ELVIS.DOC
```


A second chance: How DOS 5 "undeletes" files

By Kris Jamsa

At one time or another, nearly everyone has experienced the helpless, empty feeling that sets in the moment you realize you've accidentally deleted an important file. Because it's pretty easy to delete a critical file by accident, many companies have created undelete utilities that make it possible to undelete a just deleted file. As you've probably heard by now, Microsoft has included in DOS 5 an UNDELETE utility that lets you recover from such potentially disastrous situations.

In this article, we'll explain how file undelete utilities work, and we'll show you how to use DOS 5's UNDELETE utility to recover deleted files. However, before we get into the specifics of how undelete utilities work, let's take a look at how DOS stores files to disk.

How DOS stores files

To store files to disk, DOS allocates disk space to files in clusters (from one to eight consecutive sectors). Based on a file's size, DOS may allocate many clusters for a file. Depending on your disk's available space, the clusters may occupy consecutive disk locations, or they may spread out across your disk. DOS tracks each file's clusters using the File Allocation Table, or FAT. Figure A shows three files and their corresponding cluster chains in the FAT.

When you access a file, DOS searches the specified directory for a matching filename. A directory is simply a list containing each file's name, size, date and time stamp, and starting cluster number. Figure B illustrates a possible directory for the three files shown in Figure A.

Let's assume, for example, you want to copy the file named TWO.DAT to a diskette in drive A. When you issue the COPY command, DOS searches the directory until it finds the matching entry. Using the file's starting cluster number contained in the directory (4),

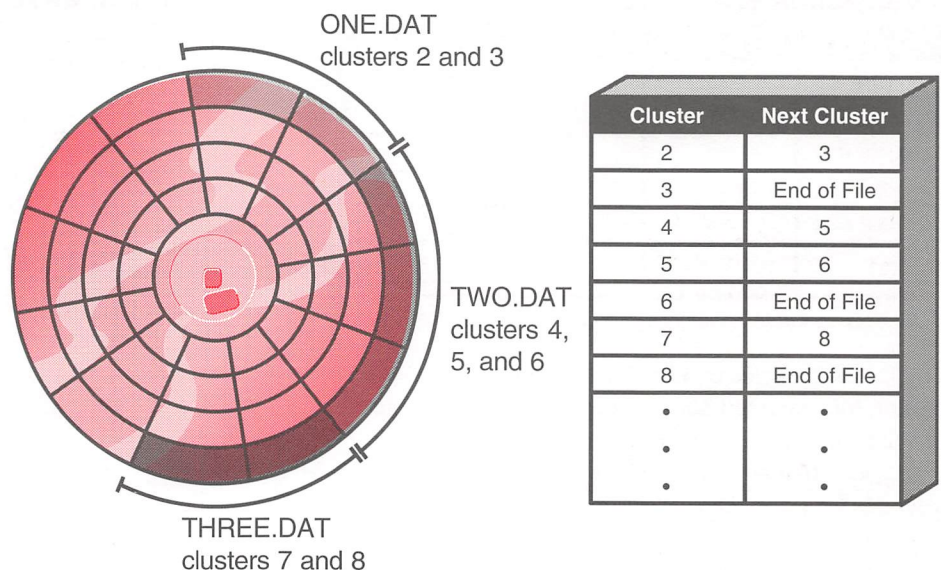
DOS can locate the file's data by following the FAT's cluster chain (4, 5, 6). Once DOS locates the file's data, it can carry out the copy operation.

What does the DEL command do?

When you delete a file from your disk using DEL, DOS really doesn't erase the file's contents from your disk. Instead, DOS performs the following two actions. First, it marks the disk clusters that contain the file as available for reuse by another file by assigning the value 0 to each cluster entry in the FAT. For example, if your diskette contains the files shown in Figure A, and you delete the file ONE.DAT, which occupies clusters 2 and 3, DOS will update the FAT, as shown in Figure C on the following page.

The second action DEL performs is replacing the first character in the file's directory entry with the extended ASCII character 0, as shown in Figure D.

Figure A



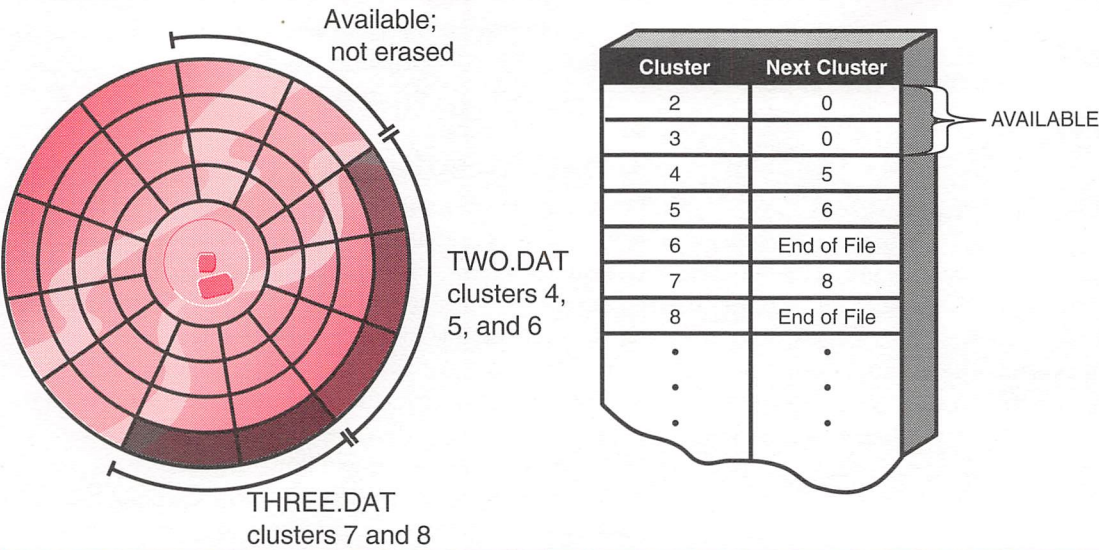
DOS uses the File Allocation Table to keep track of the files on your hard disk.

Figure B

NAME	EXTENSION	SIZE	TIME	STARTING CLUSTER
ONE	DAT	4096	12:30p	2
TWO	DAT	6144	12:31p	4
THREE	DAT	4096	12:32p	7
.		.	.	.
.		.	.	.
.		.	.	.

A directory lists each file's name, size, and date and time stamp.

Figure C



If you delete ONE.DAT from the disk shown in Figure A, DOS will update the File Allocation Table.

Figure D

NAME	EXTENSION	SIZE	TIME
ONE	DAT	4096	12:30p
TWO	DAT	6144	12:31p
THREE	DAT	4096	12:32p
.	.	.	.
.	.	.	.
.	.	.	.

DOS replaces the first character in a deleted file's directory entry.

Just as the value 0 in the FAT indicates an available disk cluster, the extended ASCII character σ in the first character of a filename indicates an available directory entry.

Notice that although DEL marks the file's clusters as available for reuse, DEL never erases the file's actual contents from disk. That explains how you can later undelete a file. However, if you delete one file and then create another, DOS may store the new file's contents in one or more of the deleted file's clusters. If this happens and DOS overwrites your deleted file's contents, you will not be able to undelete the file. Likewise, DOS may replace the directory entry of the deleted file with the entry describing the new file. If your undeleted file's directory entry is replaced in this way, most third-party undelete utility programs will not be able to undelete the file. As you'll see in a moment, however, DOS 5's UNDELETE utility can undelete a file whose directory entry has been replaced.

Undeleting files with DOS 5

Beginning with version 5, DOS provides the UNDELETE command, which "undeletes" a deleted file. When you

issue the UNDELETE command, UNDELETE searches your directory for entries beginning with the σ character. Because DOS changes the deleted file's starting character, UNDELETE will prompt you to type the correct letter.

As you will recall, the only change DEL makes to the file's directory entry is to the first character of the filename. The file's size and starting cluster number remain un-

changed. If you choose to undelete the file ONE.DAT, for example, UNDELETE will proceed as follows. First, UNDELETE determines the number of file clusters it must recover using the file and disk cluster size:

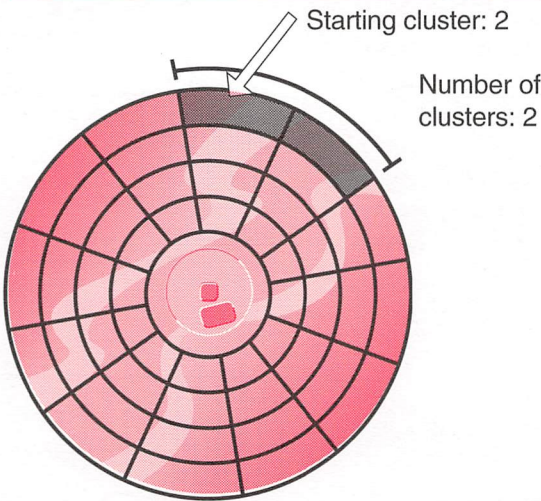
$$\text{Clusters} = \text{File size} / \text{Cluster size}$$

In the case of ONE.DAT, UNDELETE determines the number of clusters like this:

$$\text{Clusters} = 4,096 / 2,048 = 2$$

Second, using the starting cluster number from the file's directory entry, UNDELETE searches the FAT for unused clusters, recreating the original cluster chain, as shown in Figure E.

Figure E

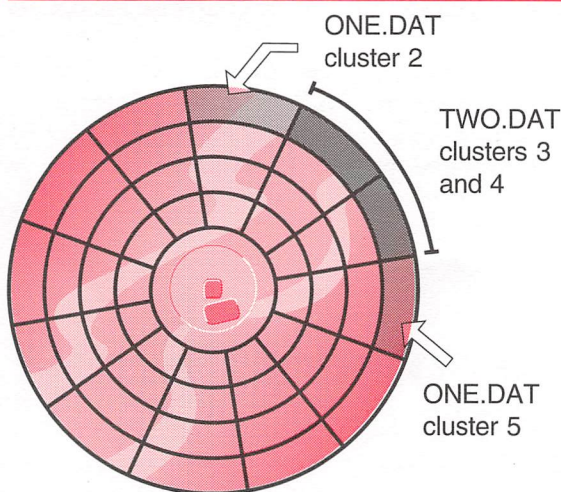


UNDELETE uses the FAT to recreate a file's original cluster chain.

Using the letter you specify for the filename's first character, UNDELETE updates the directory entry and writes the updated FAT back to disk. At that point, your file is undeleted just as you'd hoped.

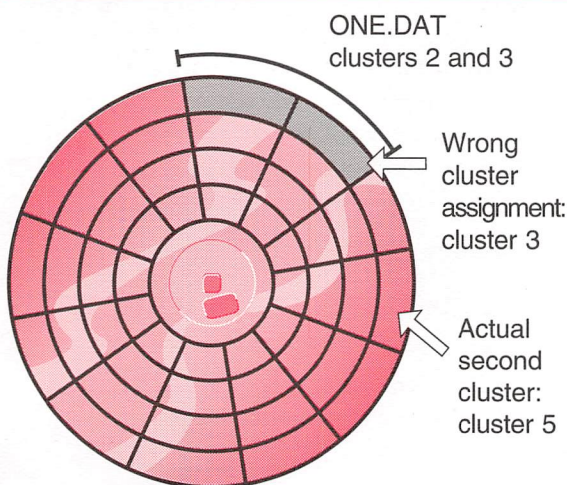
Although UNDELETE and similar third-party utilities are very powerful, they can't guarantee 100 percent recovery of your files. The major obstacle to recovering files successfully is file fragmentation. Sometimes, DOS must store files in disjointed clusters spread across your disk, as shown in Figure F. When a file's clusters become spread across your disk, the file's contents are considered to be fragmented.

Figure F



A fragmented file's contents are stored in disjointed clusters.

Figure G



UNDELETE can't always recover a fragmented file.

Suppose you delete the files ONE.DAT and TWO.DAT, shown in Figure F, and then want to undelete ONE.DAT. As we discussed, UNDELETE will first determine the number of clusters by using the file's size, which is two in

this case. Then, UNDELETE will begin at the file's starting cluster and will use the first two available clusters, as shown in Figure G.

Because the file was fragmented, its contents weren't stored in consecutive clusters. Unfortunately, UNDELETE has no way of knowing the file was originally fragmented and, as such, assigns to this file what was once one of file TWO.DAT's clusters. As a result, only a portion of the undeleted file's contents is valid. Fortunately, this type of problem occurs only with fragmented files. We'll take a look at how you can keep the number of fragmented files on your disks down to a minimum in a future issue of *Inside DOS*.

Putting the UNDELETE command into action

Let's look at a simple example of recovering a deleted file using DOS 5's UNDELETE command. Suppose the root

Protecting your disk with DOS 5's MIRROR program

By Kris Jamsa

In the article "A Second Chance: How DOS 'Undeletes' Files" beginning on page 5 of this issue, we demonstrated DOS 5's UNDELETE command. In addition to UNDELETE, DOS 5 provides a command named MIRROR that enables a process called *file deletion-tracking*. When MIRROR is active, DOS records in a file (called the deletion-tracking file) the directory entry and File Allocation Table entries for each file it deletes. Consequently, if you later need to undelete a file, UNDELETE can use the information in MIRROR's tracking file for recovering deleted, fragmented files or files whose directory entries have been used by a new file. To enable MIRROR's file deletion-tracking for drive C: simply issue the following command:

```
C:\>mirror c: /tc
```

Once you activate MIRROR, UNDELETE knows the first letter of each filename. Therefore, you don't need to specify that letter when you're undeleting files.

If you're like most DOS 5 users, you'll want to place the MIRROR command in your AUTOEXEC.BAT file so that file deletion-tracking will always be active. If you have more than one disk drive, you'll need to run the MIRROR command once for each drive, replacing the c: above with the appropriate drive letter. ■

directory on your hard disk contains these two files:

```
C:\TEST>dir
Directory of C:\TEST
.           <DIR>      05-06-91   4:41p
..          <DIR>      05-06-91   4:41p
TWO        DAT       29268 03-10-91   8:46p
ONE        DAT       31833 02-16-90   8:25a
4 file(s)   61101 bytes
196210688 bytes free
```

Additionally, suppose you use the DEL command to delete the files ONE.DAT and TWO.DAT. Now, to undelete both files, invoke UNDELETE as follows:

```
C:\TEST>undelete *.dat

Directory: C:\TEST
File Specifications: *.DAT

Deletion-tracking file not found.

MS-DOS directory contains 2 deleted files.
Of those, 2 files may be recovered.
```

Using the MS-DOS directory.

```
?WO  DAT      1  9-30-91   5:25p   ...A
Undelete (Y/N)?
```

Press *Y* and DOS responds

Please type the first character for ?WO .DAT: _

Press *T* to specify the first character in the filename TWO.DAT. DOS responds

File successfully undeleted.

and asks you if you want to undelete the file ONE.BAT. Once you've undeleted the files, DOS returns you to the prompt. ■

Kris Jamsa is the author of over 30 books on DOS, programming languages, hard disk management, Windows, and batch files. Among the books he has written are Hard Disk Power and the Jamsa Disk Utilities.

DOS BASICS

Getting to know the DOS environment

Are you environmentally aware? You probably know that when you issue the PATH command, you are telling DOS where to look for programs it can't find in the current directory. What you might not know, however, is that DOS stores your path in a special area of memory called the *environment*.

As you might expect, DOS uses the environment for much more than simply storing the path. In this article, we'll briefly introduce the environment to you and show you how to customize its contents and size to suit your personal tastes and needs.

What's stored in the environment?

To display the environment's contents, simply enter the command

```
C:\>set
```

When you do, DOS will display something like this:

```
COMSPEC=C:\COMMAND.COM
PROMPT=$P$G
PATH=C:\DOS;C:\123;C:\WINDOWS;C:\
TEMP=C:\WINDOWS\TEMP
```

As you can see, our sample environment stores the custom prompt and path we've defined, along with the system's COMSPEC. COMSPEC specifies the location of the DOS command processor, COMMAND.COM. We refer to COMSPEC, PATH, and PROMPT as *environment variables*

because the information these variables contain is stored in the environment. In addition to environment variables, the environment stores the information you provide using ANSI.SYS commands.

Using the environment

DOS isn't the only thing that uses the environment. Many application programs (including Microsoft Windows and WordPerfect) create and use their own environment variables. Similarly, many terminate and stay resident programs (TSRs) take advantage of environment variables. As you begin writing some sophisticated batch files, you'll want to look for ways to take advantage of environment variables as well.

Defining custom environment variables

To define custom environment variables, you'll use the SET command, which takes the form

```
set variable=string
```

where *variable* is the text you want to assign to the *string*. For example, if you want to create an environment variable named PRINTER, and you want to assign to that variable the text string laserjet, you use the command

```
C:\>set printer=laserjet
```


If you issue this command, then display the contents of the environment again by typing the SET command without any arguments, DOS will add to the environment the line

```
PRINTER=laserjet
```

Notice that DOS displays the variable name PRINTER in all uppercase letters even though you typed it in lowercase. You'll find that DOS always converts variable names to uppercase, regardless of what you type on the left side of the equal sign. However, notice that DOS maintains the case you specify for the text string you assign to that variable (the text to the right of the equal sign).

Once you've defined an environment variable, you can use it in a batch file's ECHO or IF command by enclosing the variable's name in percent signs (%). For instance, to display the text assigned to the variable PRINTER, you include in a batch file the command

```
echo %printer%
```

When DOS carries out this command, it will display the text you've assigned to the variable PRINTER—laserjet.

As you might guess, you can redefine an environment variable by issuing another SET command and assigning the new text to that variable.

To remove a variable from the environment, just type the SET command, followed by the variable name and an equal sign. For instance, to remove the PRINTER variable we defined a moment ago, just issue the command

```
C:\>set printer=
```

Now, when you use the SET command to display the environment's contents, you'll no longer see the variable PRINTER.

Expanding the size of the environment

The default size of the DOS environment is 160 bytes. Because DOS requires a single byte to store each character in the environment, the maximum number of characters you can store in the environment is 160. That might sound like plenty of room at first, but after you've defined COMSPEC, PROMPT, and PATH, you won't have much room remaining for storing environment variables. Fortunately, versions of DOS later than 3.3 let you expand the environment size. To do this, just insert a line into your CONFIG.SYS file that takes the form

```
shell=c:\command.com /p /e:size
```

where *size* is the desired environment size in bytes. Like all settings in CONFIG.SYS, the new environment size you specify won't take effect until you save the modified version of CONFIG.SYS and reboot your machine.

For instance, if you want to expand the size of the environment to 512 bytes, enter into your CONFIG.SYS file the line

```
shell=c:\command.com /p /e:512
```

When you save your change to disk and reboot your machine, DOS will reserve 512 bytes of memory for the environment instead of 160 bytes.

The largest environment size you can specify is 32K. Keep in mind, however, that the larger you make the environment, the less memory your machine will have available for running applications. In practice, you'll probably find that 512 bytes is plenty of environment space to suit your needs. ■

VAN WOLVERTON

Looking into the crystal ball: What lies ahead for the computer industry in 1992

Comdex—Computer Dealer Expo, the country's largest computer trade show, the annual excuse for the computer industry to trek off to Las Vegas—has come and gone for the thirteenth time. Because Comdex is the year's most important gathering of computer aficionados, the show says a lot about the state of the industry and gives me an opportunity to take a look at the year ahead.

The first Comdex trade show in 1979 was held at the MGM Grand Hotel; all the exhibits, all the exhibitors, and nearly half the attendees in one hotel. Now Comdex takes up almost two million square feet of exhibit space

scattered among the cavernous Las Vegas Convention Center, half a dozen major hotel/casinos, and the new half-million-square-foot Sands Convention Center (built by the company that sponsors Comdex, which bought the Sands Hotel and Casino, which says something about how much money they make from this affair).

This year, as for the past several years, Bill Gates and 100,000 or so of his closest personal friends took over Las Vegas for nearly a week (much to the chagrin of cabbies and casino folk who complain that computer types aren't the big spenders that Las Vegas types love. Oh well).

Despite the thousands of booths, not much new or exciting was evident in either hardware or software, but that's been the case for several years. With more than 1,500 companies exhibiting and what seems like half the state of Montana attending, it's pretty hard for a company to make much of a splash by announcing a faster 486 machine or another word processor with a few more bells and whistles.

There were some fascinating multimedia presentations by IBM, Apple, and several other companies, but most of these were either handcrafted show demos that you can't buy yet, or they used a combination of hardware and software that would cost several minutes' interest on the national debt.

Out of the clutter, however, rose possible answers to a couple of questions of more than passing interest to mere computer users like us: Who's going to build the next computer we buy? What operating system are we going to be using in a few years?

Be careful out there

The personal computer marketplace has evolved from a traditional corporate buyers' exchange where the emphasis was on image, quality, and service, to a street fight where the weapon of choice is the street price (discounted from the mythical Manufacturer's Suggested Retail Price). Prices are dropping faster than even budget-conscious buyers dared hope, and profit margins are squeezed thinner every day.

The confusing cast of hundreds of computer makers has coalesced into three broad categories: The name-brand manufacturers who charge premium prices, such as IBM, Compaq, and Hewlett-Packard; more-or-less anonymous clone makers who compete on price alone; and a clutch of relative newcomers, name-brand compatible makers such as Gateway 2000, Swan, and Northgate, who have established a reputation for price and quality and sell primarily through mail order.

The big winners are those in the last group, whose share of the market has exploded in the last couple of years. Even Dell Computers, the first mail-order PC-compatible maker to establish a reputation for quality and reliability, is under price pressure from these newer players. They're responding by moving into retail sales through computer superstores, moving them a bit closer to the name-brand category.

The Midwest strikes back

The price pressure has been excruciating as these upstart mail-order companies, many of them located far from Silicon Valley in places like Iowa and Minnesota, have signed service agreements with nationwide organizations. Now you can buy a system from them and still get local service, further diminishing the ability of the giant manufacturers to charge a premium and support their own sales and service force.

What does all this have to do with Comdex? When the show was conceived in 1979, it was pitched as a trade show

whose attendees would all be computer sellers, primarily computer stores and value-added resellers—companies, often quite small, that bought machines wholesale and added software tailored for a specific market. The personal computer industry was still in its infancy, and here was a chance for manufacturers to present their product to just the people who could make it successful.

Sure enough, Comdex was just what the doctor ordered. Alliances struck between manufacturers and resellers at those early Comdex shows played a major role in the phenomenal growth of the personal computer industry. For the first five or six years of the show, a manufacturer had to exhibit at Comdex just to establish credibility.

Stores are hurting

But computer stores have been hurt badly by the free fall in prices; if manufacturers' margins have shrunk, store margins are downright thin. Computer store personnel are no longer a representative portion of the Comdex crowd; large-volume buyers still make up the majority of attendees, but overall it's a mixed bag of corporate and government agency buyers, value-added resellers of all sizes, software developers, component manufacturers (disk drives, displays, etc.), distributors, major computer manufacturers, and other folks just interested in computers who managed to get a ticket.

The main message from the manufacturers for the dealers at the show is, "We're going to do everything we can to protect your margins." Dealers have heard this before, and are understandably skeptical. Most of the conversations quickly get around to the ultimate question: As prices continue to fall, who's going to absorb the margin loss, the maker or the dealer? There's no easy answer.

Manufacturers are next

The effect of falling prices was first felt at the retail level because the stores are the end of the distribution channel, closest to the source of the problem (we, the customers making the buying decisions). But the problems have worked their way back up the channel to the manufacturers. That's where the next round of casualties will occur, and some companies are more vulnerable than others. The troubles at Compaq—laying off nearly 1,500 employees—are probably just the beginning of a difficult period for many computer makers; Compaq is suffering first because it was most vulnerable.

Although Compaq billed its first machines as portables to distinguish them from the IBM PC (and they really were easier to lug around than the several components of a PC), from the beginning they established a reputation of quality and performance. After the rousing success of their first model, Compaq downplayed the portable aspect and emphasized performance—usually outgunning IBM in speed and storage capacity—and priced their machines either the same as IBM or even a bit higher.

Rather than simply building a clone and competing on price, Compaq worked on gaining the confidence of corporate buyers. The strategy paid off as Compaq built a loyal following and made strong inroads into the corporate market, a domain IBM preferred to think of as their home territory.

Computers are a commodity product

But Compaq and IBM have watched helplessly as both their market share and margins have shrunk in the past few years under the onslaught from the compatible makers, especially in the corporate market. Despite frequent price cuts, their machines can no longer command a significant premium over the quality compatibles. Compaq was particularly vulnerable on two counts: It doesn't have other product lines to take up the slack, as do IBM and Hewlett-Packard; and their overhead is so high they can't compete effectively in a commodity market.

IBM, too, is retrenching, but instead of laying people off, it either reassigns them or offers them early retirement. IBM talked of cutting back 10,000 employees in 1991, but will probably have to trim another 30,000 or so to get back into fighting shape while still reducing prices, coming out with more powerful machines, and reducing their manufacturing costs. IBM can do it, but it's one of the biggest challenges the company has ever faced.

Apple is vulnerable, too

Even Apple has had to respond to the relentless pressure on computer prices. For years Apple could command premium prices because Macintosh was perceived to be a superior platform for graphics, desktop publishing, and similar applications. Users were willing to pay up to twice as much as they would for an equivalent PC-compatible system.

But PC hardware and software have been gaining ground and now, for all but the most esoteric video and multimedia applications, you can do the job just as well with a PC for a lot less money. Hence the Mac Classic and the other lower-priced Apple products.

So who will make the computer we buy in a few years? There's no pat answer, but chances are good that if you bought your machine more than a couple of years ago, your new one won't be made by the same company. And if you bought your last computer from a retail store, there's a good chance that you won't buy your next one there; there are powerful reasons to buy locally, but it usually requires paying a premium and right now the premium seems too high for a lot of buyers.

What about the operating system?

The operating system question isn't so complicated. DOS is still king, but its days—well, its years—in the form as we know it are numbered. Unix is an alternative for a few, and

DR DOS is available for those who want a DOS clone, but the only real pretenders to the throne are OS/2 and Windows. (OK, Windows isn't really an operating system—it runs on top of DOS—but it offers an alternative to DOS as the main operating environment. And OS/2 requires the Presentation Manager for an equivalent operating environment; for brevity's sake, if not absolute parallelism, when I say Windows I mean the combination of DOS and Windows, and when I say OS/2 I mean the combination of OS/2 and Presentation Manager.)

A window to the future?

Microsoft was everywhere at Comdex. Besides their own booth, which was huge, noisy, and crowded, there were two separate booths for Windows applications and the pen computing version of Windows. These booths contained demonstration areas for dozens of other companies who were showing the applications they have developed for Windows. (It's rumored that Microsoft sent 500 people to Comdex; not too many years ago they didn't even have 500 employees.)

IBM, on the other hand, showed some OS/2 applications, and a few other booths on the floor had one or two OS/2 applications running, but there wasn't much excitement about OS/2. It's hasty to make too much of what you see at Comdex, but the Comdex scene does echo the evidence from elsewhere: OS/2 doesn't really seem to be catching on with the great majority of PC users. Unless IBM can do something extraordinary fairly soon, OS/2 seems destined to remain a bit player in the operating system arena, confined primarily to large corporations and government agencies who run networks with dozens or hundreds of PCs, probably linked to an IBM mainframe computer.

Windows, however, continues to gain popularity. The collapse of hardware prices has reduced the cost of the faster processor, larger memory, and larger hard disk that Windows needs. More and more applications are coming out all the time. Windows 3.1, due early in 1992, promises fewer Unrecoverable Application Errors and a better File Manager. It's difficult to see anything derailing the Windows juggernaut.

And there's a good chance that version 6.0 of DOS will look more like Windows than it does like the old familiar DOS. The command line interface, the butt of so many jokes and whines, may become, if not a thing of the past, a tool of last resort used rarely and reluctantly.

So the answer to the operating system question is that you'll most likely be using DOS for years to come, but it will probably look and feel like Windows. So go out and get that mouse. You're going to need it sooner or later. ■

Contributing editor Van Wolverton is the author of the best-selling books Running MS-DOS 5 and Supercharging MS-DOS. Van, who has worked for IBM and Intel, currently lives in Alberton, Montana.

MS-DOS 5 Technical Support

(206) 646-5104

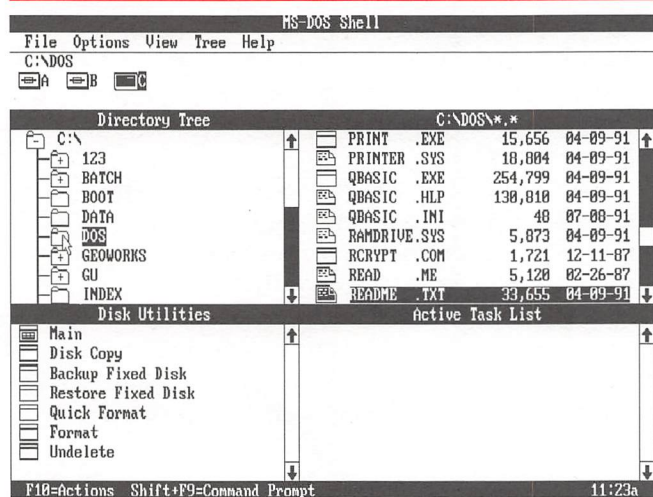
Please include account number from label with any correspondence.

Continued from page 1

Viewing the contents of a text file in DOS 5's Shell

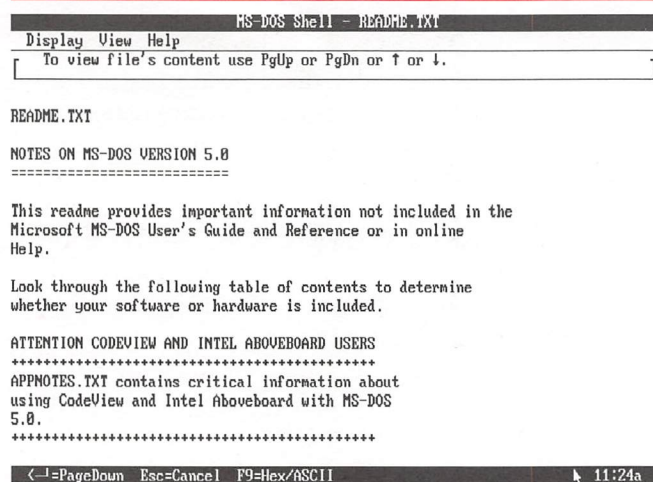
When you press **R**, DOS will highlight the first file in the directory beginning with the letter **R**. To move the highlight to **README.TXT**, just press the **↓** key until the filename scrolls into view, as shown in Figure B.

Figure B



When you select a filename and press **[F9]**...

Figure C



...DOS will display the file's contents.

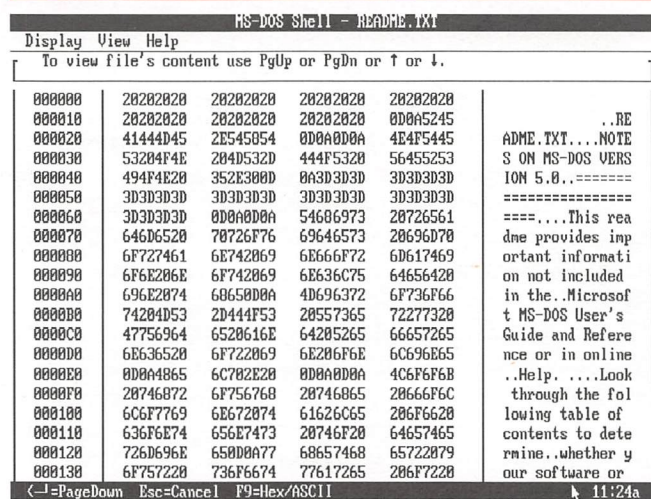
Once you've selected **README.TXT**, pull down the File menu and issue the View File Contents command, or

press **[F9]**. DOS will display the contents of the file, as shown in Figure C.

To scroll through the text file, press the **[Page Up]**, **[Page Down]**, **↓**, or **↑** keys. When you're finished reading the file, you can return to the Shell window by pressing **[Esc]**.

Notice in Figure C that the View window's status bar offers an option called **F9=Hex/ASCII**. This means that by toggling the **[F9]** key in the Shell you can view the contents of the file in either plain text or hexadecimal code. The hexadecimal code represents the binary numbers used by your computer. You can take a look at **README.TXT**'s hexadecimal code by pressing the **[F9]** key, as shown in Figure D.

Figure D



By toggling the **[F9]** key in the Shell, you can look at a file's underlying code.

(By the way, to view the **README.TXT** file at the prompt, you should move to the **\DOS** directory and issue the command

C:\DOS>type readme.txt | more

You need to include the **MORE** command whenever you view a file that contains more than one screen of text. Appending **MORE** to the **TYPE** command tells DOS to display the file one screen at a time.)